Editor: Jim X. Chen, jchen@cs.gmu.edu

RAPID PROTOTYPING OF VIRTUAL ENVIRONMENTS

By Marco Lanzagorta, Robert Rosenberg, Lawrence J. Rosenblum, and Eddy Y. Kuo

THE GOAL OF VIRTUAL REALITY IS TO FULLY OR PARTIALLY IMMERSE A HUMAN IN A VISUALLY COUPLED ENVIRONMENT.[1] BY TRACKING THE POSITION AND ORIENTATION OF THE USER WITH SENSORS DESIGNED FOR THIS PURPOSE AND BY COUPLING THESE measurements with a high-performance computer graphics system, we can generate a computer-synthesized view of a *virtual environment* that responds to the user's movements. Thus, the user does not just see a visual display on a terminal but is immersed within the display. VR also allows natural real-time interaction with the VE. Instead of a GUI, the system uses perceptual and multimodal interfaces (such as gesture, audio, and speech recognition) to interact with the data. Natural locomotion devices let the user navigate through the VE. Also, because of the size of the typical projection-based VR display (CAVEs and workbenches), groups of scientists and engineers can more easily work together to interpret data, making full use of the 3D portrayal.

Scientific visualization is the process of displaying scientific data with computer graphics in a way that facilitates understanding and analysis. Because large-scale computational efforts usually result in complex, multidimensional data sets, data analysis and exploration can greatly benefit from VR's natural 3D interface. VR scientific visualization is simply an interface between the researcher and the computer, integrating the advances made over the last decade in both scientific visualization and VR. This interface lets scientists interact with their data in the same fashion that they interact with real-world objects. In many scientific endeavors, however, there is no reality to the data. Life-sized molecules or miniaturized field lines of a solar prominence are not real objects but are useful representations of a scientific process. However, interacting with the data using natural interfaces enhances the process of scientific understanding.

The 1999 Supercomputing conference, with over 20 VR displays on the exhibition floor, demonstrated the proliferation of VR systems to examine data from large-scale computational problems. However, VR is still far from being a general research tool. A VR application usually requires large investments of time and expertise in computer graphics, so scientists prefer to rely on desktop visualization tools.

At Naval Research Laboratory, we have been developing advanced visualization software that eases the transition between desktop and VR visualization. Our aim has been to reduce the work and time needed to port a desktop visualization to a VE. Our software is a general system for rapidly prototyping VR visualizations, and users do not require any knowledge of VR. Once researchers are satisfied with the desktop visualization, they can port it to the VE in a very short period of time—ranging from a few minutes to a few hours, depending on the complexity of the visualization and on the size of the data sets.

VR technology overview

VR systems provide the user with a sense of being immersed in the data, where objects have a sense of "presence." To achieve immersion, we require both *scene involvement* and *object persistence*. Scene involvement means placing the user in a sufficiently large display area so that the real world does not impinge on the computer-generated images, thus establishing presence. Object persistence means that the viewer has the feeling that an object will still be there, that it will still exist, even when he or she looks away. If the system tracks the user's head, the graphics computer redraws the scene according to the user's new viewpoint. VR systems require interactive frame rates of at least 20 Hz to maintain this illusion.

To provide presence, a typical VR setup for scientific visualization must include a high-speed graphics computer that can render at least five million polygons per second, a tracking device to sense the user's head position, a device to interact with the VE, and a VR display. We limit our discussion here to those VR components that are useful for scientific visualization. Visit http://vl.nrl.navy.mil/staff/lnzgrt/grotto/index.html for further information about VR and SV and for additional Web resources.

Acronym glossary

Application Visualization System	AVS
Cathode-ray tube	CRT
Graphical room for orientation, training, and tactical observation	Grotto
Head-mounted display	HMD
Level of detail	LOD
Liquid-crystal display	LCD
Naval Research Laboratory	NRL
Projection-based display	PBD
Virtual environment	VE
Virtual reality	VR
Virtual reality modeling language	VRML

VR displays. The two most popular VR display technologies are head-mounted displays and projection-based displays. Developed in the 1990s, PBDs have become the primary VR display technology and are credited with VR's movement outside the laboratory and into industry.

HMDs consist of a set of goggles mounted on a helmet with small monitors that generate images for each eye. Monitors are usually liquid-crystal displays or cathode-ray tubes. CRTs provide higher resolution but are heavy; LCDs are lighter but have lower resolution. Both types are combined with head tracking for complete immersion. A variant of this is the binocular-orientation monitor, which consists of two 1,280 × 960 color pixel CRTs that are counterweighted on a free-standing platform (either floor or desktop) with a mechanical link for six degrees of freedom motion.

PBDs range from large single-wall configurations to multiwalled immersive rooms. As the name implies, PBDs employ high-resolution projectors (1,024 × 768 pixels) with large fixed screens set at a distance from the viewer, along with shutter glasses for stereo viewing. These systems provide high-resolution images in a stable display, giving an unencumbered VE that multiple viewers can share. Because of the shared VE, PBDs have become very popular for demonstrations—placing a burden on many visualization labs around the world, aptly named "the curse of the Immersadesk."

Stereo viewing is readily available with almost all SGI systems by using Crystal Eyes stereographic shutter glasses. The stereo viewing system comes with one or more infrared emitters that have a 10- to 15-foot signal range, which we can conveniently place almost anywhere around the workbench or the immersive room. A user then dons a pair of goggles with an infrared receiver at the nose bridge. The lenses are nematic liquid crystals, which darken when oriented with an electric field. The two lenses alternate at 120 Hz. Synchronization with the stereo video mode on the SGI platform using the infrared sensors assures that the system displays the left eye image whenever it darkens the right lens and vice versa. Perception of stereo viewing not only facilitates understanding the 3D morphology, it is essential to simultaneously perceiving near and far field images.

The one wall configuration is the least immersive and is primarily used as a high-resolution display for large audiences. Multiple SGI Onyx pipes render a single image, each pipe controlling a single projector generating panoramic images of up to 3,200 × 2,400 pixels. Interaction is limited in most of these systems because the screen is typically front-projected, and direct interaction would block the projector's image.

Workbench systems use a high-resolution projection system, a mirror, and a table with an imaging surface. The VE is created on a high-performance graphics computer and projected to a mirror and then toward a table's imaging surface. The imaging surface's tilt angle is adjustable. Horizontal surfaces are best suited for tabletop tasks, including many scientific visualization tasks such as medical and terrain visualization. Figure 1 shows a medical application of the workbench system.

Immersive rooms are surround-view, surround-sound, multiperson theaters, typically sized approximately 10 × 10 × 10 feet (see Figure 2). The system projects stereographic images onto multiple walls (three in the earliest configurations, four to six is now typical). Rendering real-time, stereoscopic images on multiple walls requires very high-end graphics performance. Thus, the graphics computer accounts for most of the cost of these systems. Fortunately, VR costs will decrease as PCs with graphics accelerators and large texture map memories begin to perform VR tasks that previously required high-end workstations.

Trackers. The most commonly used trackers are electromagnetic. They consist of a stationary emitter and one or more receivers. Both the emitter and the receivers have three perpendicular coils that completely determine position and orientation. Depending on the system, the orientation is specified as either a 3 × 3 rotation matrix (nine values), quaternions (four values), or Euler angles (three values). Because Euler angles require only three specified values, they provide fast update rates. However, Euler angles suffer from singularities at the north and south orientation poles. Quaternions have no such singularities and are the format of choice when available.

Interaction systems. The most common interaction device for workbenches and immersive rooms is a *3D wand*. Typically, this is a joystick with a tracking sensor attached to provide orientation and position information. The VR software receives and processes the tracking data, essentially providing a simple form of gesture recognition. Buttons on the joystick help users navigate through a VE as well as select orientation.

A second interaction device is a glove

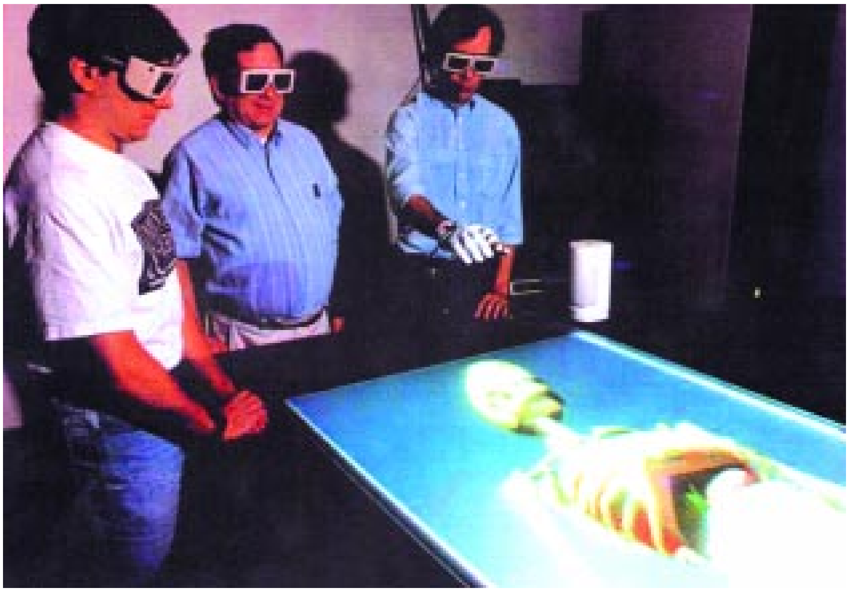

Figure 1. The Naval Research Laboratory produced the first workbench system in the US in 1994. Since then, many applications have been ported to this virtual reality display, including this medical application.

with a tracking sensor attached. One form of the glove has conducting plates attached at the tips of each glove finger. When the user presses the two plates together by touching fingertips, he or she triggers an event. Based on these simple pinch gestures, we can program a variety of actions into the VR application. More complex gloves contain the ability to measure finger movement and provide tactile and haptic feedback.

Researchers have also used voice-recognition software to interact when simple wand or hand gestures are not natural or just not feasible. For example, in a medical visualization where users are examining a human skeleton, they can select, move, and manipulate individual bones independent of the whole skeleton. It is nearly impossible to reattach the bone in the original position through simple glove gestures. When the user issues the simple verbal phrase "put back," the voice-recognition software can return the skeleton to its original configuration no matter how dismembered it became during analysis.

VR software. VR software has many components to support real-time VE generation. Some of these components are *interaction*, *navigation*, and *modeling software*. Interaction software provides the mechanism to construct a dialogue from the various control devices and to apply that dialogue to a system or application to change the display accordingly. This is a critical component of VR systems that involves both hardware and software. Visual scene-navigation software provides the means to move the user through the 3D virtual world (controlling the viewpoint and direction in the VE). Additionally, it should support *level of detail*, which is critical for efficient scene navigation through polygonally dense data sets. LOD is based on multiresolution methods where the system displays an object with more polygons as the viewer approaches the object and fewer as the viewer recedes from it. Modeling software defines the form, behavior, and appearance of the objects in the VE.

Rapid-prototyping software

To meet the needs of NRL's scientific community, we developed advanced visualization software that lets scientists rapidily prototype VR applications on their desktop, without large investments of time and knowledge of computer graphics. The resulting visualization is readily accessible for VR visualization and can accommodate a variety of interaction and display devices.

Our software is based on two libraries: the SGI IRIS Performer and NRL's VRLib. Performer is a high-performance rendering library that SGI developed and optimized to handle extremely large and complicated graphical environments. It supports a rich range of input and output data formats. In addition, the library supports multiple processors and multirendering hardware. These technical requirements are necessary for a system such as an immersive room, where the system must project a synchronized image onto four screens at interactive rates. NRL developed the VRLib libraries to harness the Performer libraries for VR application development. They provide a set of configuration and interaction capabilities across a range of display and interaction devices. VRLib handles the rendering, interaction, navigation, and head tracking in the VE. In immersive rooms, the libraries also handle the image synchronization across the different screens.

Our software consists of two main applications: the SciVis Viewer and the AVS Viewer. They differ in the degree of interactivity they provide and the kind of data set they can visualize.

The SciVis Viewer displays in stereo any Performer-supported-format model with VR navigation and manipulation. The user can then view and analyze the model in the VE using rotations, translations, scaling, or navigation through the model, in real time. This visualization type is very useful when the object is time-independent and has distinct boundaries, such as isosurfaces of static scalar fields, molecular protein structures, and grain boundaries within a metallic alloy.

Time-dependent data has proven most popular with the SciVis Viewer. It provides an interactive environment for viewing 3D animations of the data. To prepare the animation, the user generates a sequence of virtual reality modeling language files for each simulation time step. Using software developed at NRL, we converted the

Figure 2. This Naval Research Laboratory Grotto (graphical room for orientation, training, and tactical observation) is an immersive-room display. The image above is a material microstructure. NRL is developing several applications for this VR display, including scientific and battlefield visualization.

VRML files into a Performer-based animation file. We can then view the animation in the VE just as any other VRML model. As the animation repeats, we can change the view point at will to examine dynamic structures from any conceivable angle. When combined with the immersive room's large-scale VE, this provides a compelling visual analysis that is not possible from any other display.

Of course, the animation is from precomputed data, and there is no interactivity aside from straightforward 3D object manipulation. For interactivity, we use a more sophisticated software package: the AVS VR Viewer.

Many commercial visualization systems support additional functionality for increased interactivity (such as changing a colormap, an isosurface value, or the lengths of a set of stream lines), increased data-exploration capabilities (subsampling and selecting regions of interest), and real-time interactive visualization (visualizing data during the simulation, modifying the simulation, and seeing the effects in real time). Because all these functionalities are highly desirable for a VR scientific visualization system, we built a VR interface to AVS (Application Visualization System), one of the most popular software packages for scientific visualization that the NRL scientific community uses.

Like the other commercial visualization systems, AVS is a reliable, user-friendly program that lets users create visualization applications by combining software components into executable networks. The components, called *modules*, implement specific visualization functions. These functions can filter, map, or render. These packages also let users create their own modules to meet their specific needs and to dynamically load them into networks. They can run these new modules on a remote platform, such as high-performance computers, allowing high-performance scientific visualization.

The AVS VR Viewer module is an AVS wrapper for the SciVis Viewer. This module replaces the geometry viewer module of the visualization networks and displays information on any VR display the VRLib libraries support. Because the rendered image is part of a visualization network, we can interactively perform all the visualization functions permitted by the visualization system module library and the user's handwritten modules.

The module also permits texture-based volume rendering using the OpenGL volumizer library. The OpenGL volumizer is a library based on OpenGL that uses a 3D texturing technique to simulate a volume-rendered image. By adding the library to the AVS VR Viewer, we added the volume-rendering capability to our software. Because we are using a 3D texture technique, we can easily combine additional geometries from other AVS modules with the volume rendering in the same scene.

The advantages of such a system are clear. Scientists do not have to be expert programmers or make a large time investment to visualize scientific information in a VE. They just develop at the visualization system level. As part of a visualization network, only the performance of the machines used for the computation and visualization processes constrains the resources available. Also, scientists can develop their visualization networks at their desktop without going to the VR laboratory.

Case studies

Recently, we supported NRL scientists in more than 25 scientific visualization projects. These projects cover a wide range of scientific disciplines including materials sciences, chemistry, biochemistry, computational fluid dynamics, and space sciences. Each of the two following projects has its own unique visualization aspects, given the data-analysis requirements.

Material science and technology. The influence of the internal microscopic material structures, or *microstructures*, on the mechanical properties of materials is well established. To fully understand the interaction between structure and properties, materials scientists must comprehend a microstructure's 3D shapes. This case study describes a technique for the 3D analysis of microstructures of materials.[2] The technique consists of incrementally polishing through a thin layer (approximately 0.2 μm) of material, chemically etching the polished surface, applying reference marks, and performing optical or scanning-electron microscopy on selected areas. We then process the series of images to obtain a 3D reconstruction of the material.

The 3D reconstruction is done by digitally modifying each image to improve the contrast. Next, we convert the image stack from all the sections into a volume data set (a 3D scalar

Marco Lanzagorta is a senior systems engineer working for Scientific and Engineering Solutions. He is developing scientific visualization applications for the Naval Research Laboratory's Immersive Virtual Environment. His research interests are computer graphics, virtual reality, and augmented reality as well as their scientific and defense applications. He received his BS in physics from the University of Mexico and his PhD in physics from the University of Oxford. He is a member of the IEEE Computer Society and the National Defense Industrial Association. Contact him at the NRL, Code 5594, 4555 Overlook Ave. SW, Washington, D.C. 20375; lnzgrt@volt.nrl.navy.mil; ohm.nrl.navy.mil/staff/lnzgrt.

Robert Rosenberg is a computational scientist in the Center for Computational Science at the NRL. His scientific visualization research interests include real-time interactive visualization, computational steering, visualization of large-scale scientific data sets, and virtual reality. He received his BA in chemistry and mathematics from the University of Rochester and his MS, MPhil, and PhD in theoretical physical chemistry from Columbia University. Contact him at the NRL, Code 5594, 4555 Overlook Ave. SW, Washington, D.C. 20375; robert.rosenberg@nrl.navy.mil; vl.nrl.navy.mil.

Lawrence J. Rosenblum is director of virtual reality systems and research in the Information Technology Division at the NRL. His research interests include VR, scientific visualization, and human–computer interfaces. He has published over 70 papers and has edited two books, including *Scientific Visualization: Advances & Challenges* (Academic Press, 1994). He received his PhD in mathematics from Ohio State University. He is on the *IEEE Computer Graphics and Applications* and *Virtual Reality* editorial boards and on the *IEEE Transactions on Visualization and Computer Graphics* advisory board. He is a former chairman of the IEEE Technical Committee on Computer Graphics. He is an IEEE senior member and a member of the IEEE Computer Society, ACM, Siggraph, and the American Geophysical Society. Contact him at the NRL, Code 5580, 4555 Overlook Ave. SW, Washington D.C. 20375; rosenblu@ait.nrl.navy.mil; www.ait.nrl.navy/vrlab.

Eddy Y. Kuo is a senior software engineer at Autometric. (The research reported in this paper was performed while he was a member of the NRL VR Lab.) His research interests include computational geometry, physical-based modeling, real-time simulation, interactive visualization, and VR applications. He received his BS in mathematics from Pittsburgh State University and his MS from Ohio State University. He is an IEEE Computer Society and Mathematical Association of America member. Contact him at Autometric, 7700 Boston Blvd., Springfield, VA 22153; ekuo@autometric.com.

field) and generate an isosurface. This isosurface represents the 3D reconstruction of the microstructure. In particular, we apply this technique to an alloy steel sample using a stack of 250 images. The 3D reconstruction shows microstructural features not previously identified with traditional 2D techniques on single planes of polish.

During the study, we explored the visualization of these microstructures in an immersive room. Because this is a time-independent object with very clear boundaries, the SciVis Viewer was the best software for the VR visualization. Thus, the first step was to create the VRML model of the microstructure. To have a smooth model, as well as the best possible resolution in the VR display, we performed a tricubic interpolation of the original 3D data set that quadrupled the number of grid points. Using this data set, we generated an isosurface and then transformed it into a VRML file. Most of the visualization systems available output VRML files.

The resulting VRML file was time-consuming to render due to the large number of triangles involved (nearly one million). To achieve a semblance of real-time interactivity, the system required a minimum frame rate (usually 20 frames per second). To this end, we limited the model's triangle count to 30,000. (To render an object in stereo, we must compensate for both the left and right eye images by doubling the number of rendered triangles and then multiplying by the number of screens where the rendering will take place [four in immersive room]. Our hardware—a six-processor ONYX 2 SGI computer—can only handle five million triangles per second.) We used *triangle decimation* techniques to maintain real-time rates.

Triangle decimation is a 3D data-compression technique for surfaces represented as triangle meshes. These techniques reduce the total number of triangles in a triangle mesh, preserve the original topology, and form a good approximation to the original geometry. We use them to improve rendering at interactive rates for large objects. For this case, we discovered that the best way to optimize the triangle decimation routines, while preserving the desired level of model detail, was to use two or more routines in a nested, hierarchical fashion. In this way, we managed to reduce the total number of rendered triangles by a factor of 30.

The SciVis Viewer let us inspect the microstructures both as a whole and then as separate components. The 3D effect produced and the size of the images displayed gave the user a real sense of the microstructures' morphology and their spatial distribution that normal computer monitor screens cannot match (see Figure 2). During one of the VR visualization sessions, the principal scientist, George Spanos, observed a new kind of structure. Although scientists could conceivably observe such structures on the computer monitor screen, they did not noticed them in numerous analysis sessions until they viewed the data in VR inside the immersive room. This demonstrates that VR's capabilities can produce new understanding of data sets previously studied with conventional techniques.

Computational fluid dynamics. This computational fluid dynamics application numerically simulates an axially-excited 3D free square jet.[3] For the purposes of visualization, we computed a reduced model using a grid size of $40 \times 60 \times 40$ so that we could interactively explore the simulation's parameter space for an optimal jet engine design. This is a perfect example of combining high-performance computing and high-bandwidth networks with VR environments.

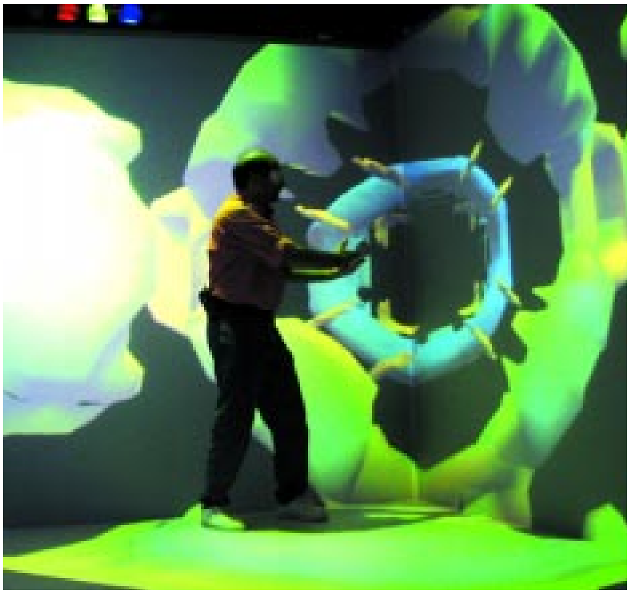

Figure 3. The VR visualization lets scientists stand inside the jet plume.

During the simulation, a sinusoidal forcing function injects fuel into the computational domain through a square jet nozzle. The computational domain is initially filled with oxygen and the vorticity surfaces indicate the amount and type of mixing that occurs as the fuel travels downstream from the nozzle. Combustion is not included in the simulation. The visualization shows large vortex rings, which indicate large-scale transport and mixing of reagents as well as smaller braid vortices, indicating small-scale mixing. Both types of mixing are required for efficient combustion. As the simulation progresses, we can modify various simulation parameters such as the frequency and amplitude of the sinusoidal forcing function as well as the relative velocities of the fuel and oxygen.

We converted the simulation program into an AVS module that runs on a high-performance computer such as a 16-processor SGI C90 or a 32-processor SGI Origin 2000; the AVS network runs at the VR graphics workstation. A high-bandwidth ATM network connects both machines. Each time step, we calculate and download the vorticity of the vector field to the graphics workstation where we render the vorticity isosurfaces in a VR display (using the AVS VR Viewer module). For example, the user can step into the Grotto and be surrounded by the isosurface. With the AVS VR Viewer module, the user can fully interact with the visualization by changing the calculation and the visualization parameters (for example, the physical quantity being visualized, the isosurfaces' level, or the color map).

With dedicated time on a C90, the simulation runs interactively—the bottleneck occurs in the rendering of the isosurfaces. Unfortunately, it is rare to get dedicated time on high-performance computers for interactive visualization.

In response to these problems, we created a Performer-based animation. We generated VRML files of the isosurfaces for every 10th time step for the first 400 time steps of the simulation. We then produced a Performer-based cyclic animation, and the VR visualization is straightforward using the SciViz Viewer. Although there is no interaction with the animation, except for 3D manipulation, the impact of the visualization was very high. Users in an immersive room could actually stand inside the jet plume and observe the large vortex rings flowing around and rising above them (see Figure 3).

Virtual reality offers many advantages for scientific visualization: stereoscopic rendering, high-resolution images, wide-angle viewing, and a natural human–computer interaction interface. VR seems to be the optimal visualization method for problems where the data set's 3D structure and topology are scientifically relevant. Our main effort has been to develop software that eases the transition between desktop visualization and VR visualization. We are still working on the AVS Viewer so users can manipulate all the visulation parameters from inside the VE. Our intent is to develop visualization tools that researchers can apply in a wide range of scientific areas without spending too much time in software development. As a result, we ported approximately 25 applications to a VE in about one year.

Although many problems remain to be solved (especially those arising from the large data-set visualization), VR seems to be an excellent technology for visualizing complex 3D data sets, and it might play a dominant role in future scientific research. We also feel that VR shows great potential as a device for communicating and disseminating of visual information.

Our software is freely available to universities and research institutions by contacting the authors.

Acknowledgments

This research was supported by the Office of Naval Research and the Naval Research Laboratory. We thank the NRL researchers who provided us with interesting data sets for visualization: George Spanos and Milo Kral for the materials science data and Fernando Grinstein for the computational fluid dynamics data.

References

1. National Research Council, *Virtual Reality: Scientific and Technological Challenges*, National Academy Press, 1995.
2. M. Lanzagorta et al., "Three Dimensional Visualization of Microstructures," *Proc. IEEE Visualization*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1998.
3. U. Obeysekare et al., "Virtual Workbench: A Non-Immersive Virtual Environment for Visualizing and Interacting with 3D Objects for Scientific Visualization," *Proc. IEEE Visualization*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1996.